Enhancing On-line Customer Trust:

Animated Advisor Design and Prototype

by

Rami Musa

Submitted to the Department of Electrical Engineering and Computer Science

in Partial Fulfillment of the Requirements for the Degree of

Master of Engineering in Electrical Engineering and Computer Science

at the Massachusetts Institute of Technology

May 14, 2003

[June 2003]

Author
Department of Electrical Engineering and Computer Science
May 14, 2003

Certified by
Glen L. Urban
Thesis Supervisor
David Austin Professor Of Marketing
Management

Accepted by
[illegible]thur C. Smith
[illegible]aduate Theses

Enhancing On-line Customer Trust:
Animated Advisor Design and Prototype

by

Rami Musa

Submitted to the

Department of Electrical Engineering and Computer Science

May 14, 2003

In Partial Fulfillment of the Requirements for the Degree of

Master of Engineering in Electrical Engineering and Computer Science

Abstract

The emergence of e-commerce has significantly increased customer power and introduced many challenges to the traditional push-based marketing practices. Many businesses nowadays are considering repositioning their strategies to develop trust-based long-lasting relationships with an increasingly loyal customer base instead of bombarding passive customers with marketing campaigns. General Motors took a leading first step in this direction by launching an on-line tool, Auto-Choice-Advisor, that provides its visitors with unbiased advice on the vehicles that best fit their preferences.

This document proposes trust-building enhancements to the Auto-Choice-Advisor site through the incorporation of a trusted animated advisor. It begins by reviewing the premise of trust-based marketing. It then presents simulation results that internally validate the "listening-in" methodology, an approach that listens to virtual dialogues between on-line customers and virtual advisors to identify viable new market opportunities. The document then builds on the findings and presents a design and a prototype for a new animated Auto-Choice-Advisor. Finally, the possibilities for future work are summarized.

Thesis Supervisor: Glen L. Urban

Title: David Austin Professor Of Marketing, MIT Sloan School of Management

Acknowledgements

First of all, I would like to thank Professor Glen Urban for his valuable advice and guidance and for giving me the opportunity to work with him on this exciting project over the past year. I gained a great deal of exposure to marketing principles while building on my technical engineering background. I am grateful for his patience and the confidence he had in me. I highly appreciate the opportunity he gave me to present my work to General Motors management at the company headquarters in Detroit, MI.

I also would like to thank the team members: Professor Fareena Sultan, Philip Decamp, Stanley Cheung, Iakov Bart, Lauren McCain, Mahesh Kumar and Sandra Crawford-Jenkins for the enjoyable working experience with which they provided me. Special thanks to Philip Decamp for his assistance in running simulations and recording the animation audio.

I would like to thank my mother, father and siblings for their love and support. Special thanks to my sister Amal for her guidance and advice always. Many thanks to my many friends for their support and for lending an ear.

Finally, many thanks to my high school advisor, late Mr. Sarkis Melidossian, for his support and confidence in me.

Contents

Chapter 1: Introduction

1.1 Background

1.1.1 Trust-Based Marketing

Trust-based marketing is an approach in marketing that nurtures and deepens the relationship between a company and its customers. In contrast to the traditional "push" marketing approach in which passive customers are bombarded with marketing campaigns, a trust based marketing strategy builds a stronger relationship with an increasingly loyal customer base. While traditionally the "push" approach assumed that customers did not know what is best for them, this is now changing.

Customer power is growing [1]. Customers nowadays have greater access to information. The Internet provides customers with the means to find out for themselves what best suits their needs. Therefore, customers are increasingly able to evade marketers' often biased push messages. In addition, customers now have access to more options. With the spread of e-commerce, customers are no longer reliant on local physical stores and agencies. The simplified transactions that the Internet enabled have made it easy for customers to switch between different venders or service providers independent of their physical locations. This dramatically widened the range of competition for businesses. Moreover, several on-line customer communities have evolved around interest in particular products and services. Customers nowadays form forums on-line to share information and stories about both their good experiences and their bad ones. This can have great impact on businesses specially in competitive saturated markets where the demand falls short of the supply.

A trust-worthy business that advocates its customers' long-term interests can benefit a lot through the trust marketing strategy. Maintaining a trusting customer base would help reduce the costs of acquiring new customers [1]. If customers trust a company they are less likely to switch to other venders or service providers. Therefore, the company would not need to acquire as many customers to maintain its target growth level and hence it can save on its marketing campaigns. Similarly, customers are likely to tolerate higher prices from businesses they trust than from business they do not trust [1]. Therefore, a trust worthy company can enjoy higher profit margins. A trusted company can also play a leading role in its market since it has a closer relationship to its customers and therefore has better access to information about customer needs [1].

In order for a company to incorporate a trust strategy, it needs to satisfy certain prerequisites. For example, a prerequisite to gaining customer trust is that a company must provide high quality trustable products [1]. If that is not the case, then a trust-based relationship cannot be built. Furthermore, several changes need to be introduced to implement a trust strategy. A company needs to become more transparent to its customers [1]. Customers need to be able to see through the processes of the firm. They need to be able to track their orders, address any processing issues and have deeper access to information about the company and the products it provides. In addition, a company needs to align itself on the customer side by fixing and avoiding any conflicts of interest that might exist between the firm and the customer as well as between different customers [1]. A company would also need to provide its customers with unbiased information and advice as well as comparisons of its own products against those of its competitors [1].

A trust-based marketing strategy is not the right model for all companies.

There are several characteristics about the nature of a business that determine whether trust is the way to go [1]. Some are discussed below:

1. *The competitive environment:* On one side of the spectrum, if the market in which the business operates is highly competitive with many substitutable competing products, then there will probably be little benefit to developing a closer relationship with the customers since they are less likely to turn into to loyal customers. On the other side of the spectrum, if a business is a monopoly then the business may not benefit much from a trust strategy either since it already enjoys high power over its market and hence the prices.
2. *The operating environment:* It is important for companies that implement a trust-based strategy to have the ability to maintain a high and consistent quality of services and products. Moreover, they need to have the ability to adapt to changing levels of demand without any disruptions in service and without jeopardizing quality.
3. *The nature of the customer and the product or service:* Building a long-term trust relationship with customers naturally implies that customers engage in a long-term relationship with the company. If customers only engage in short-term deals with the company or buy products that they would not use on a regular basis for a long time, a trust strategy is not likely to be beneficial to the company.

1.1.2 The Auto-Choice-Advisor Site

General Motors (GM) is one company that is positioned well to consider integrating trust into its strategy and operations. GM is a leading car manufacturer in

North America. It has the financial and operational capabilities to provide high quality products and to adapt its production levels to handle changes in demand in a timely fashion. Furthermore, vehicle purchase is a long-term engagement that involves the customers with a product that they use on a regular, often daily, basis. Therefore, GM enjoys the market positioning, the operational and financial ability and the customer characteristics that would enable it to build a long trust-based relationship with its customers. GM has been studying ways to build on trust in its different marketing channels.

AutoChoiceAdvisor.com is a website created by GM as part of its trust effort [1][6]. The site assists its visitors in identifying the vehicles that best fit their preferences. The visitors answer questions about their preferences and can at any point during their visit view the recommendations of the site. The recommendations are

Figure 1.1: AutoChoiceAdvisor.com, a General Motors Sponsored Site

completely unbiased and, in fact, the recommendations might not include any GM vehicles. The website derives its data from AIC, the Automotive Information Center, and J.D. Power & Associates. Figure 1.1 shows the Auto-Choice-Advisor welcome page.

In addition to helping build a trust relationship with customers, the Auto-Choice-Advisor site provides a fast way to collect market information provided by customers in a more honest setting. Customers on the site reply to questions about their preferences while knowing that the site is unbiased. Therefore, they are likely to provide answers that well reflect their preferences.

1.2 Project Plan

This research project has two phases. The first phase is to wrap up the work for a previous project, called Trucktown, by performing a final internal validation check on a new methodology for identifying unmet customer needs. The methodology was designed by professors Glen Urban and John Hauser at the MIT Sloan School of Management [4]. It involves listening in to virtual communications between on-line customers and trusted virtual agents in which customers answer questions about their preferences and the products for which they are searching. The collected data is later analyzed to detect any unsatisfiable requirements, unmet needs, and finally viable market opportunities are evaluated through a clustering analysis.

The project then begins its second phase. The goal then is to design a new trusted virtual advisor for the GM-sponsored website, AutoChoiceAdvisor.com based on the findings from the Trucktown project as well as other trust-based marketing research. The new virtual advisor is to be designed, prototyped and finally presented to GM executives to explore ways of incorporating the new on-line trust findings into a potential multi-channel trust marketing strategy for GM.

1.3 Team Contributions

Previous and current team members contributed to this project as follows:

Team Member's Name	Contribution
Stanley Cheung and Iakov Bart	Clustering analysis on collected Trucktown data and initial error sensitivity testing.
Philip Decamp	Static HTML talking-head demo for Auto-Choice-Advisor.
Rami Musa	Error and trigger level sensitivity tests, dynamic JSP talking-head demo, prototype of question answering and virtual engineer units for Auto-Choice-Advisor.

1.4 Organization

This document is organized into six chapters. Chapter 2 presents an overview of research in Internet trust-based marketing and the findings that would be useful in the design and implementation of a new version of the Auto-Choice-Advisor site. Chapter 3 summarizes the findings from a series of sensitivity tests on the "listening in" methodology for identifying customer segments with unmet needs. "Listening in" can add value to the market data collection and processing performed on the data collected from Auto-Choice-Advisor. Chapter 4 overviews the virtual advisors currently deployed on the Internet and sets the specifications and goals for the prototype of the future Auto-Choice-Advisor website. Chapter 5 describes the design and the implementation of the Auto-Choice-Advisor prototype. Finally, chapter six summarizes the goals and results of the project and addresses the potential ways to proceed forward.

Chapter 2: Trust Research Overview

2.1 Trucktown: Identifying Unmet Customer Needs

Trucktown is a research project that was led by professors Glen Urban and John Hauser at the MIT Sloan School of Management [2][4][5]. Trucktown is an on-line tool that provides unbiased trustable recommendations to its users about the trucks that best fit their preferences. Trucktown's virtual advisor engages in dialogues with the visitors, collects data about their preferences and then tries to match them to what is available in the market. When the visitor's requirements cannot be met by what is in the market, a requirement conflict is said to exist and at the end of the visit a virtual engineer asks the visitors further questions about their needs that caused the conflict. The Trucktown project also involves a clustering analysis on collected conflict data to identify economically viable opportunities in the truck market, any unmet customer needs. Below is an overview of the different modules of the Trucktown project.

2.1.1 The Virtual Advisor

The virtual advisor component of Trucktown converses with the visitors and collects information about their preferences regarding truck specifications. The advisor has a database of priori probabilities for each truck in the market that it uses to calculate a utility value for each truck. The utility value indicates the extent to which a truck fits the requirements specified by the visitor.

The advisor asks two types of questions. At the beginning, a constant-sum preference question is asked and is used to determine the initial utility value associated with each truck. Further questions are in a nominal category format where the user selects, on a fixed range such as 1 to 5, the extent to which they care about a particular attribute of a truck. The nominal category questions are ordered based on a 2-step Bayesian look-ahead that determines the next question that would provide the most information. After answering each question a new utility value is calculated for each truck and the truck that provides the maximum utility is noted. At the end of the visit the truck with the highest utility is recommended to the visitor.

2.1.2 The Virtual Engineer

When a visitor answers a new question, the maximum utility, the utility of the best matching truck, should increase if the requirement the user specified when answering the question can be met. If it cannot, then the maximum utility drops. The virtual engineer module "listens in" to the dialogues between the visitor and the advisor and tries to detect drops in utility. At the end of a visit, if a drop in utility higher than a trigger level is detected, the virtual engineer is launched. The engineer gathers information from the user targeted towards understanding the customer needs behind the requirement that caused the utility drop. For example, if the user specified that he or she needs a compact truck that can tow, then the utility drops since compact trucks cannot tow. Figure 2.1 below illustrates an example of a drop in maximum utility.

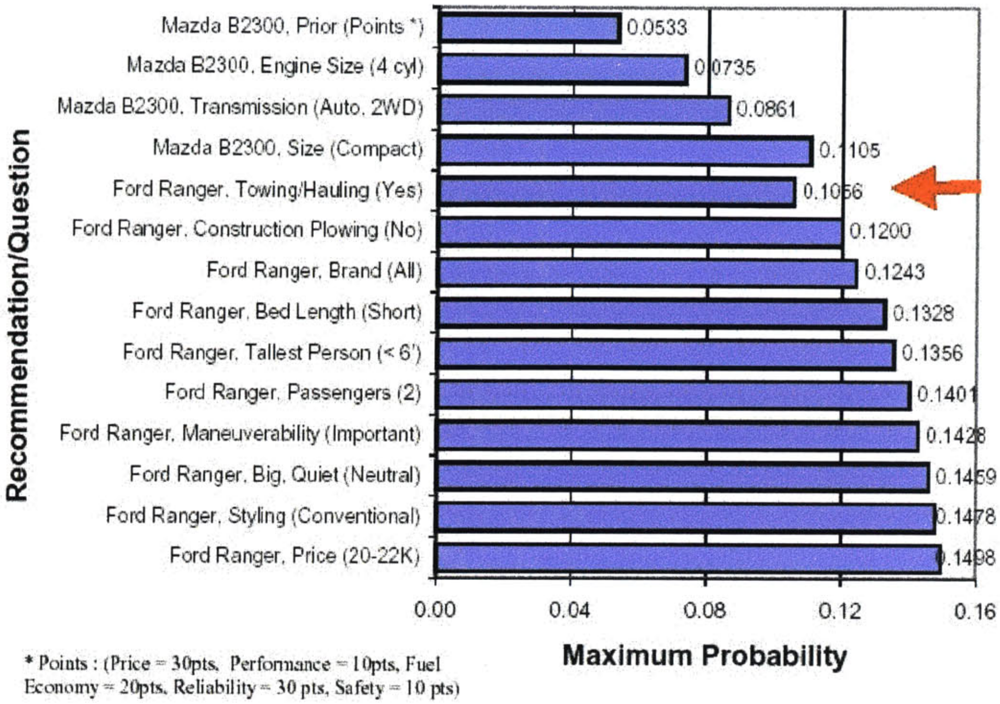

Figure 2.1: Drop in Maximum Utility [2]

2.1.3 The Design Pallet

The design pallet functionality is a way to engage Trucktown users in a more personable way. The users can specify the attributes that they want in their truck and the model of the truck and how it would look is presented. When the user introduces modifications that change how the model looks, the model is updated immediately.

2.1.4 The Listening In Methodology: Identifying Unmet Needs

At the end of the virtual advisor session, if a utility drop occurred, an engine on the back-end examines every pair of questions the visitor answered and tries to identify which pair or pairs of questions caused the drop. For this purpose, the system maintains, in the priori probability database, records of the conditional probabilities for each truck given each possible question-answer combination. The correlations between answers provide a good indication of which pairs of answers correspond to a conflict. The research team decided that any correlation less than -0.3 would indicate a conflict among the corresponding answers. Following this definition of a conflict, the Trucktown project included 208 possible conflicts. When a utility drop is detected, a conflict log for the visit is recorded and it indicates which of the 208 conflicts are present.

In order to check for any sizeable market segments that have unmet needs, the Trucktown research team performed a clustering analysis on the conflict data they had gathered. Any sizeable cluster of visitors with similar conflicts would indicate the presence of a customer segment with unmet needs that might represent a viable market opportunity. Chapter 3 will cover in further detail the "listening in" methodology and will present the results of an internal validation check of the methodology's success.

2.2 Website Characteristics as Determinants of Trust

In order to design a new animated trusted advisor for the Auto-Choice-Advisor site, it is important to review the research findings about the factors that drive trust on the web. This section reviews some major findings that are important to take into consideration.

Marketing research indicates that several website characteristics affect customers' trust perceptions on a website. Those include privacy, security, presence of trust seals, ease of navigation, brand, advice, absence of errors, presentation, order fulfillment and community. Urban, Sultan, Shankar and Bart [3] conducted a study in which they found that while privacy and security characteristics have traditionally been important in determining users' likelihood to trust and use a website, they contribute to only about 11% of the variance in the perceived website trust. The researchers found that, as the Internet matured, other factors seem to have become more prominent drivers of trust on-line. In particular, navigation, advice and brand seem to play a bigger role in driving website trust than perceived privacy and security.

Furthermore, Urban, Sultan, Shankar and Bart [3] find that trust is a mediating factor in explaining the behaviors and actions of website users. Therefore, trust plays a major role in determining how successful ebusiness websites are in stimulating the desired behaviors and actions by the customers and thus in serving the goals for which they are created. Chapter 4 will make use of these research findings when it proposes some trust-building additions to the Auto-Choice-Advisor site.

Chapter 3: The "Listening-in" Methodology: Monte Carlo Simulations

3.1 Introduction

Section 2.1 summarized a practical marketing methodology that showed how major unmet market needs can be identified by "listening-in" to dialogues between on-line virtual advisors and customers. Analyzing user responses to questions about their preferences and identifying any conflicts, requirements that cannot be satisfied by the products in the market, can be the basis for a clustering analysis that can group users with similar conflicts together, thus revealing any major customer groups with the same unmet needs.

The listening in methodology's ability to identify unmet need segments might be sensitive to the conflict trigger parameter, which in the case of Trucktown determines whether the virtual engineer is launched, as well as user response errors. It is almost impossible that users give the same answers even when their needs are the same. Users might interpret questions differently or they might simply mistype their responses. This motivated performing Monte Carlo simulations on simulated Trucktown data to study the effects of different trigger levels and different response error levels on the performance of the clustering analysis. This chapter documents the simulation process, results and findings.

3.2 Simulation Methodology:

3.2.1 Simulation-Generated Samples

The simulation process first involves generating a sample of user responses on which to perform the clustering analysis. Nine profiles, describing the preferences and choices of nine different customer segments, are defined. Six of the chosen customer segments have preferences and choices that cannot be satisfied by trucks currently available in the market. The remaining three customer segments have no unsatisfiable preferences or choices. A simulation program is then used to generate 500 user responses per profile, a total sample of 4,500 simulated users.

The Trucktown virtual advisor program is then simulated with each of the generated user responses. As was described in section 2.1.4, a conflict log is recorded for each user. The log contains an entry for each of the 208 possible conflicts and indicates whether the conflict exists in the user's responses. In absence of response errors, users with no unmet needs have conflict logs that do not contain any conflicts. In contrast, users with unmet needs have responses that cause conflicts. If a conflict causes a utility drop that exceeds the trigger level, the conflict log is updated to indicate the presence of all the conflicts that exist in the user's response. Finally, the conflict logs for all 4,500 simulated users are pooled together and used in clustering simulations that attempt to identify the unmet-need groups in the sample.

The nine customer segment profiles used are described below:

Segment Profile	Truck Preferences and Desired Features
Profile 1:	Compact truck that can tow and haul.
Profile 2:	Full-size 8-cylinder truck with a short bed
Profile 3:	Conventionally-styled compact truck with diesel engine
Profile 4:	Compact truck with an extra short bed, a tall driver
Profile 5:	Compact truck with a short bed and a 10-cylinder engine
Profile 6:	A truck with a long bed and high maneuverability

Profile 7:	A compact truck with no additional features
Profile 8:	A full-size truck with no additional features
Profile 9:	A full-size truck that can tow and haul

A detailed description of the profiles is available in appendix I.

3.2.2 The Clustering Procedure

A program, SPSS, is used to run a k-means non-tree clustering algorithm on the conflict data arranged in a matrix of the 208 possible conflicts and the 4,500 simulated users. The clustering algorithm groups users with similar conflicts together. To run the algorithm, the first step is to specify the conflicts on which to base the clustering and the number of clusters that the algorithm should generate. The program is then run. The output is examined to check whether the identified clusters correspond to the original customer segment profiles. The clusters' sizes and their conflict composition can be used to evaluate the algorithm's ability to identify the corresponding unmet need segments.

The conflicts on which the clustering simulations are based include all possible response conflicts except for those that are related to specific manufacturer brands and truck body styles. The reason is that users are not likely to be aware of what individual truck manufacturers offer. Similarly, they are unlikely to be able to differentiate between different truck styles based on style names. Therefore, user responses to brand and style questions are not likely to be true reflections of the users' preferences.

In order to choose the number of clusters the program should generate, the simulation is run several times. The first run specifies seven output clusters. Seven is the minimum number of clusters expected in the data; one cluster for each of the six profiles with conflicts due to unmet needs in addition to one cluster combining the three conflict-free profiles. The output clusters are then examined to check whether they have interpretable meanings. Each cluster is checked to see whether it corresponds to any of the six conflict profiles or is the conflict-free cluster. The simulation is then repeated, incrementing the number of clusters each time, until the seven expected clusters are reflected in the clustering output and any extra clusters are too small to have any interpretable meaning.

After performing the iterations, using seven to thirteen clusters, nine was identified as the best number to use. The corresponding clustering output included the seven expected clusters and two small clusters with no valuable meaning. Nine was subsequently used in the sensitivity tests.

3.2.3 Trigger levels and Response Errors

Three series of clustering simulations were performed. The first aimed to check whether the listening in methodology is able to identify unmet needs in presence of moderate levels of error in user responses. The second was to study the sensitivity of the methodology to the trigger level and check the robustness of the chosen trigger. The third, and final, series of simulations was to study the performance of the methodology in presence of different response error levels.

Adding Error To User Responses:

Trucktown users responded to two types of questions: constant-sum self-explicated importance questions and questions with nominal categories. Errors in self-explicated importances were generated by adding zero-mean, normally distributed error such that the standard deviation of the error equals a specified number of points. After adding the error, any negative self-explicated importances were truncated to zero. In case of the nominal categories questions, a percentage error was used. For example, with 10% error, the user is 90% likely to answer a particular question correctly. For the remaining 10% the user is equally likely to give any of the alternative responses.

The first simulations to test the ability of the listening in methodology to identify unmet needs used 5 point standard deviation error for self-explicated importances and 10% response error for nominal categories. Error sensitivity tests included 0, 5 and 10 point standard deviation errors for constant-sum questions and 0%, 10% and 20% errors for nominal categories questions. A summary of the different simulation runs and the corresponding error levels is tabulated below. Section 3.5 presents and discusses the corresponding results.

	0% Error	10% Error	20% Error
0-Point Error	Run A	Run B	Run C
5-Point Error	Run D	Run E	Run F
10-Point Error	Run G	Run H	Run I

Table 3.1: Error Sensitivity: Simulations Summary

Trigger Levels:

The listening in methodology relies on clustering conflict data that is recorded only if the response conflicts cause a utility drop that exceeds the trigger level.

Therefore, it is important to verify that the trigger level used is sensitive enough to detect important utility drops associated with unmet needs. For this purpose, the methodology was tested with all trigger levels ranging from 0 to 0.0001, with 0.000025 increments in addition to trigger levels equal to 0.00015, 0.001, 0.01 and 0.1.

3.3 Initial Clustering Simulations

The initial clustering simulation used a moderate level of response error (5-point constant sum error and 10% nominal category error) and a trigger level of 0.00005. In total, 82.73% of the users were correctly assigned to clusters that match their original customer profiles despite the presence of errors. The clustering results are summarized in table 3.2 below.

	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Total
Profile 1	**418**	0	0	1	0	0	81	0	0	500
Profile 2	1	**422**	0	0	0	0	77	0	0	500
Profile 3	0	0	**401**	0	0	0	99	0	0	500
Profile 4	1	0	0	**346**	0	0	153	0	0	500
Profile 5	3	27	0	0	**336**	0	134	0	0	500
Profile 6	0	2	0	0	0	**346**	92	43	17	500
7, 8 & 9	43	0	2	1	0	0	**1454**	0	0	1500
Rate	**82.73%**									

Table 3.2: Initial Clustering Simulation Results

The entries in the table indicate the number of users from a customer profile that were matched to an output cluster. In other words, each column in the table represents one of the output clusters and the distribution of its members across the original profiles. For example, going down the first column shows that Cluster 1 is composed of four hundred and eighteen users from profile 1, one user from each of profiles 2 and 4, three users from profile 5 and forty three users from the conflict-free profiles. Similarly, each row in the table represents a customer segment profile and the assignment of its members to the output clusters.

The tabulated results show that the listening in methodology succeeded in identifying all of the true customer segments and correctly classified 82.73% of the users. The numbers in bold indicate the largest number of users from each customer segment that were grouped together. As shown in the table, each output cluster is composed mostly of users that share the same customer profile. Furthermore, the majority of users with the same customer profile are assigned to the same output cluster. This shows that the clustering algorithm has been able to generate one major cluster for each of the true customer segments. The extra two clusters, clusters 8 and 9, are too small to have any interpretable meaning. A further check examined the conflict composition of each output cluster and verified that each of the major clusters indeed corresponded to one of the true customer segment profiles (see appendix I, part 4).

There were some misclassifications. For example, while four hundred and eighteen of profile 1's users were correctly grouped together, one user was assigned to cluster 4 and eighty one users were assigned to the conflict-free cluster, cluster 7. These misclassifications were due to the presence of response errors. In the case of profile 1, eighty one of its members had response errors that changed their conflict profile to one that is close to the conflict-free profiles. One member's response errors changed the conflict profile to one that is close to profile 4. Response errors can change the conflict profile by introducing changes that eliminate any large enough utility drop, in which case the trigger mechanism is not launched and no conflicts are detected. Alternatively, the trigger mechanism might still be launched but some expected conflicts might not be present and other unexpected conflicts might be introduced due to different responses caused by the errors, effectively changing the user's profile.

3.4 Sensitivity to the Trigger Mechanism

The trigger sensitivity simulations results , summarized in table 3.3 blow, suggest that 0.00005 is a sensitive and robust enough trigger level.

Trigger Level	Percentage of Users Classified Correctly	Percentage of Opportunities Identified	Percentage of Unmet-Need Segments Identified	False Opportunities Identified
T = 0.000000	82.73%	100%	100%	0
T = 0.000025	82.73%	100%	100%	0
T = 0.000050	82.73%	100%	100%	0
T = 0.000075	82.73%	100%	100%	0
T = 0.000100	82.69%	100%	100%	0
T = 0.000150	82.69%	100%	100%	0
T = 0.001000	82.71%	100%	100%	0
T = 0.010000	56.69%	63.60%	63.40%	0
T = 0.100000	33.33%	0.00%	0.00%	0

Table 3.3: Trigger Sensitivity Simulations

Roughly the same results are obtained when the trigger value is in the range of 0.00000 to 0.00100. About 82.7% of users are correctly classified in clusters that match their customer segment profiles. All market opportunities, conflicts due to unmet-needs, are identified. All of the customer segments are identified by the clustering algorithm and no false opportunities are found.

Increasing the trigger level further introduces a drop in performance due to the less sensitive triggers. At trigger level 0.01000, only 56.69% of the users are classified correctly. Similarly, only 63.6% of the opportunities, conflicts due to unmet needs, and only 63.4% of the customer segments with unmet needs are identified. At trigger level 0.1, none of the market opportunities are identified and neither are any of unmet-need customer segments. The only users that are correctly classified are those who belong to the conflict-free customer segments, the segments with no unmet-needs.

For all trigger levels, no false opportunities were ever identified. This was determined by checking whether any of the output clusters contained an unexpected conflict for more than 50% of its members. This was never the case. In most cases, unexpected conflicts appeared for less than 20% of a cluster's members. A summery of this check is available in appendix I.

3.5 Sensitivity to the Level of Response Errors

Tables 3.4 (a), (b) and (c) below summarize the results from the errors sensitivity runs.

	0% Error	10% Error	20% Error
0-Point Error	99.93%	82.87%	61.56%
5-Point Error	99.87%	82.73%	55.00%
10-Point Error	99.91%	81.80%	56.78%

(a): User Classification Rates

	0% Error	10% Error	20% Error
0-Point Error	100%	100%	93.94%
5-Point Error	100%	100%	75.76%
10-Point Error	100%	100%	81.82%

(b): Percentage of Opportunities Identified

	0% Error	10% Error	20% Error
0-Point Error	100%	100%	100%
5-Point Error	100%	100%	63.40%
10-Point Error	100%	100%	63.40%

(c): Percentage of Unmet Needs Segments Identified

Table 3.4: Error Sensitivity Runs

The results show that in absence of response errors practically all users are classified correctly into clusters that match their profiles, all customer segments are identified and so are all the market opportunities, the conflicts due to unmet-needs.

Increasing the response errors for the constant-sum questions has little effect on the classification of the users and the identification of segments and opportunities. For example, with 10% error in nominal category questions, a 10-point error in the constant-sum questions reduces the user classification success by a mere 1.07% (from 82.87% for 0-point error to 81.8% for 10-point error). The limited effects of errors in constant-sum responses are due to the fact that constant-sum questions are used at the start of the a Trucktown visit to set the priori probabilities which are modified as the respondent answers the nominal category questions which follow. Therefore, the effects of the constant-sum questions diminish as they are over-shadowed the nominal-category questions.

At 20% errors in the nominal-category responses, the identification of segments and opportunities drops. With 5-point and 10-point error in the constant-sum responses, the algorithm fails to identify the entire customer segment represented by profile 2 (Full-size 8-cylinder truck with a short bed). Moreover, with 0-point and 10-point error in constant sum questions, the algorithm fails to identify two opportunities, conflicts due to unmet-needs, in the customer segment described by profile 1. These are the 4 cylinder engine with towing and the 4 cylinder engine with hauling conflicts.

The effects of errors in the nominal-category responses are evident in the user classification results as well. For example, at 5-point constant-sum response error, the algorithm's ability to correctly classify users drops from 99.87% with 0% error to about 82.73% with 10% error. The performance drops even more dramatically to 55% with 20% error due to fact that the entire profile 2 customer segment is no longer identifiable.

It is unlikely that customers make mistakes in 20%, 1 out of 5, of the nominal-category questions. This error level is likely to be too severe to be reflective of how the listening in methodology would perform in practice. A moderate level of error such as 10% is more reasonable to expect. At such a moderate error level, the results show that the methodology is able to classify users correctly at a rate of about 82.7%. It is also able to identify all true market opportunities and true unmet-need customer segments. It never identifies any false opportunities or segments.

Finally, the data in table 3.4 includes two surprising results. First, the classification result in absence of any errors is not 100% but 99.93%. As can be seen from the output of run A in appendix I, the SPSS clustering algorithm misclassifies 3 uses that it places in two extra clusters, a cluster with 1 user and a cluster with 2 users. This seems to be due to the program's way of assigning clusters. It seems that since 9 clusters were requested at the start, the program assigns the data points to clusters such that exactly 9 clusters are generated and such that each cluster contains at least 1 user. This also seems to be the case for runs D and G where the nominal category error is 0%.

The second surprising result is that run I generates a higher classification rate (56.78%) than run F (55%) even though run I has a higher constant-sum error and the same nominal-category error level. Both of these runs use 20% nominal-category response error. Run F uses 5-point constant-sum error while run I uses 10-point constant sum error. The most likely explanation is that the error distributions for the 5-point and the 10-point errors are not simply normal distributions that differ merely by the standard deviation value. The truncation to zero of any negative value of the self-explicated importances in effect changes the distributions. At 10-point error, the standard deviation is higher than with 5-point error (10 versus 5) and so the self-explicated importances are more likely to fall into the negative range with 10-point error than with 5-point error. Therefore, for the 10-point error case, a higher percentage of the simulated users have self-explicated preferences that are truncated to zero. Therefore, not only do the two distributions differ in their standard deviation of error before truncating negative values to zero, but they also have substantially different occurrence frequencies for the zero value. In effect, this difference seems to change the initial priori probabilities in a way that helps the algorithm better identify the user groups in run I.

3.6 Summary and Conclusion

The simulation analysis presented in this chapter showed that the Listening in methodology is robust and can be a very useful approach. The results summarized in tables 3.2 through 3.4 show a reasonable level of validity of the approach and the findings that can be derived through it. While customer classification drops when errors are introduced, the opportunities and unmet-need segments are still identified successfully at reasonable error levels which makes the Listening in methodology a great resource for new product development.

With the increasing use of the Internet for making purchases and seeking information, the listening in approach can benefit a wide range of markets including the travel, vehicle, entertainment, software and even health services markets.

Chapter 4: Enhancing the Auto-Choice-Advisor Tool

4.1 Benchmarking of Current On-line Virtual Advisors

Several approaches to design and implement on-line virtual advisors have been attempted. Current advisors are built with several functionalities that aim to make it easy for customers to find what they need, solve any problems they might have and complete on-line purchases as fast and as conveniently as possible. This section presents an overview of the features of already deployed on-line advisors and assesses the approaches that can be used in designing the next Auto-Choice-Advisor.

4.1.1 Determining Customer Preferences

When helping on-line customers find or determine the products that suit them best, the customer is often asked to input information about their preferences via a simple on-line form. Most commonly, the customer is asked to select the attributes that they like to have in their products or alternatively eliminate those that they do not like. Sometimes the customer is asked to rate, on a fixed-range scale such as 1 to 5, how much they care about a particular attribute. In order to assist the user in understanding some of the questions that require background knowledge, some advisors provide explanations to educate the user.

The responses a customer provides are then often fed into an engine, embedded in the virtual advisor, that tries to model the customer's preferences. The aim is to use the customer's input to calibrate a model that predicts what the customer is looking for. The virtual advisor can then proceed and interact with the customer according to the predicted preferences. The advisor might present products that are likely to suit the customer's needs or make recommendations and ask questions accordingly.

A common method to determine customer preferences is to use an expected utility model. The expected utility is modeled on product or market knowledge and calibrated using the customer's responses. This approach was used in the Trucktown project and is increasingly popular. A similar approach is to use conjoint analysis in which the customer's preferences are determined by presenting the customer with a series of products and asking the customer to indicate which ones they prefer. In addition, some virtual advisors use collaborative filtering to guess customer preferences. These advisors produce personal recommendations by identifying similarities between one customer's responses and those of other customers.

Table 4.1 below lists the common mechanisms and approaches discussed earlier together with some examples of virtual advisors that use them.

	Experion	Amazon	Deal Time	Kelley's Blue Book	Vehix	Auto Choice Advisor
Product Comparison	√		√	√	√	√
Attribute Selection	√		√	√	√	√
Attribute Selection & Education	√		√			
Collaborative Filtering		√		√		
Full Conjoint						
Expected Utility	√			√		√
Expert/Customer Review		√		√		

Table 4.1: Surveying Some On-line Advisors

4.1.2 Engaging the Customer

Some companies have developed creative ways to engage their website visitors and increase their affinity to the site. Below are some of these ideas:

Animation

The use of animation and specially the animation of human talking heads can have powerful effects on engaging website visitors. The combination of several human senses, visual, auditory and potentially speech, in the interaction with the site can speed up the visitors' comfort with the site and willingness to use it or even return to it in the future. While a few websites have started using animated advisors the technology is just emerging. Several technology providers already exist. This includes LifeFX, Famous3D, Pulse3D and Anthropics.

Visitor-Customized Experience

Some sites engage visitors by providing them with a more personal experience that goes beyond the common personalization and customization approaches of my preferred links and customized weather forecasts. For example, the Trucktown project included a design pallet which allows the tool's users to design the truck they want and the requirements they specify are reflected immediately in the visual display. Lands' End's website provides its customers with a virtual model, named "my model", which is a functionality that allows the user to create a virtual character with the same general body features as the user. The user can then dress up the personal model with the clothes he or she is considering to purchase.

Live Chat and On-line Forums

Some sites provide access to customer service representatives and technical experts online though chat rooms, on-line forums and on-line appointments setup via electronic mail. The examples are many. General Motors currently has a customer support mechanism which enables customers to contact customer service via e-mail and arrange an appointment with a representative. Many computer technology firms such as Dell an Sun sponsor on-line expert forums where their customers, who are generally highly technically savvy, can report any problems they might face and seek help from fellow customers or from company experts who would join in at these forums. This approach enables fast feedback from the customer community about product performance and problems as well as fast unveiling of any new market needs.

Virtual Engineer

The virtual engineer used in the Trucktown project engages the visitors by asking for further information about their preferences with the understanding that their contribution will help the development of better products in the future. The engineer is triggered when the user specifies requirements that cannot be met by any one product in the truck market. The virtual engineer then engages the site visitor in further dialogue in which it gathers specific information related to the detected unmet need potentially to be used in future product development.

4.2 Auto-Choice-Advisor: Where it Stands and Where it Can Go

4.2.1 The Current GM Auto-Choice-Advisor

The Auto-Choice-Advisor site now runs in two versions, an HTML version and a Macromedia Flash version, both serving the same functionalities. The site enables visitors to specify the requirements they need to have in their vehicles using the attribute selection survey approach. At any point in time the visitor can preview the advisor's vehicle recommendations based on the information the visitor had input that far in time. The visitor can select vehicles among the recommendations and compare them.

Several trust builder approaches are already setup. Most importantly the site provides the visitors with unbiased recommendations. While it is a GM site, it is possible that all of the top recommended cars be non-GM vehicles. The site also displays trust seals that certify the visitors' security and privacy on the site. Additionally, the Auto-Choice-Advisor uses attribute selection with education where users can learn more about what they are being asked prior to giving their answers. Moreover, the Auto-Choice-Advisor site always provides a recommendation based on what it learnt about the visitor's preferences. The visitors are never faced with a situation where no cars can be found to meet their needs. A set of best matches is always provided given any partial information the user provided. This avoids users leaving the site with any sense of dissatisfaction which would be detrimental to trust. The Auto-Choice-Advisor also provides the visitors with a dedicated personal "garage" space on the recommendation pages where they can add the cars they like and refer to them at any time.

4.2.2 Recommended Trust Building Enhancements

The Auto-Choice-Advisor as it stands achieves a lot in gaining customer trust due to its unbiased recommendations, easy navigation, privacy protection, brand and the other trust building features mentioned earlier. The main recommendation to take trust on the Auto-Choice-Advisor site to the next level is to add animated virtual agents. Human animations can add a lot to the trust experience on the Internet. Adding animations of human talking heads and enabling users to enter-act with them is the best approximation to real-life customer-agent interaction that can be deployed on the Internet. The envisioned main functionalities of the animated agent are outlined below:

Guiding On-line Visitors through AutoChoiceAdvisor.com

A virtual advisor would play the role of a human agent or advisor. In its simplest capabilities, an animated advisor can speak to the site visitors welcoming to the site. It can then walk them through the survey questions, speak out to them any information that would be helpful in clarifying what questions mean. It can also point out the site's functionalities, where they are located on the site and how they could be used. Figure 4.1 below shows a screenshot of an animated advisor that was added to a prototype of the future Auto-Choice-Advisor site. His name is Louis and he uses human face gestures, together with special site effects, to draw the users' attention to the site navigation buttons and how they be used.

Figure 4.1: The Animated Talking Head Louis

Answering Customer Questions

An even higher added value to an animated advisor is to use it to enable more life-like dialogues. For example, in the context of answering customer questions an animated advisor could accept questions from the site's visitors, in a somewhat defined context, and provide the answers on the fly. Natural language processing and other artificial intelligence technologies could be used to power the back-end of the animation. A back-end engine would analyze the user question, try to understand it, find the relevant answer or construct an English language response and finally let animation speak it to the user.

There are multiple levels of questions that the animation might be designed to handle, all with varying levels of difficulty. The easiest scenario would be answering questions that merely seek data or simple pieces of information. This includes questions about vehicle specifications, prices and warranty deals for example. A next level of sophistication would be to power the animation with the analytical ability to perform comparisons between vehicles. This could be, for example, to find the vehicle with the highest towing capacity from the group of recommended vehicles or even identifying the distinguishing specifications of a particular vehicle from among the recommendations. Figure 4.2 below shows an example of Louis on the recommendation page displaying a question that the user might have and the response that Louis would give.

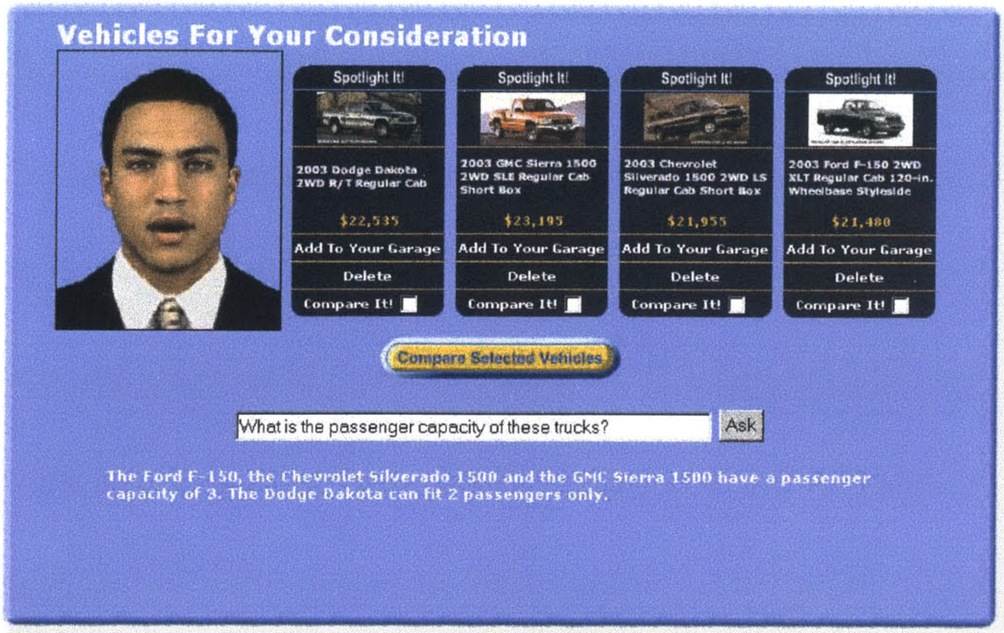

Figure 4.2: Question Answering

Virtual Engineer Triggered By Unsatisfiable Requirements

As mentioned earlier, the Auto-Choice-Advisor site processes customer responses to the site's survey questions and uses a utility estimation to identify the vehicles that best fit their preferences. This enables fast data collection about the market and the customers. A further enhancement to this data collection mechanism is to deploy a virtual engineer that is triggered when a visitor specifies requirements that cannot be met by the vehicle available in the market. The engineer would enter-act with the visitor and invite her to provide more specific information about her needs that can be used in future vehicle design as well as in better identifying the vehicles that best suit her needs. This enables the Auto-Choice-Advisor site to collect more specific pieces of information about the unmet requirements and the reasons customers specify them. This would be more relevant and on the target information that can be used in new product design.

Additionally, engaging the customer and making her feel her feedback is valuable can enhance her trust of the site. The customer might feel part of a design process that cares about their personal needs and feedback. Figure 4.3 below shows part of an envisioned dialogue with the virtual engineer, personalized by the same persona Louis.

Figure 4.3: Auto-Choice-Advisor: The Virtual Engineer

4.2.3 Usability and User-Friendliness Considerations

Several usability and user-friendliness considerations have to be taken into account when deploying virtual animated advisors online since many could affect the user's experience with the site as well as the extent they trust it.

Animations are somewhat bandwidth demanding. While several new animation technologies with lower bandwidth requirements are evolving, it still remains an important usability concern specially that many Internet user still have low bandwidth dialup connections. This calls for the need to alert website users, upon arrival on a site containing animations, about the bandwidth requirements and to provide them with the option of viewing an animation-free version of the site.

Browser compatibility is also a concern. Many web technologies are browser and version specific. Therefore, website designers need to detect what browser the customer is using to visit the site and if it is not compatible with the animation technology, an animation-free site would need to be served to the user preferably unnoticeably.

Some Internet users prefer noiseless text browsing. Many Internet users revert to the Internet to find information they need quickly. Often they need noiseless browsing without any sound, and hence without a talking head animation. Therefore, it would be necessary to provide the customers with the ability to turn off the animation, silence it and replay it later if necessary.

Chapter 5: Animated Advisor: Prototype Design and Implementation

A first demo of the improved Auto-Choice-Advisor site was built by team member Philip Decamp. This first demo was built as static HTML pages and included an animated talking head, Louis, that walks website visitors through the site and points out the functionalities of the site as well as clarifies what some technical terms mean. Therefore, the first demo implemented the first envisioned enhancement to the Auto-Choice-Advisor site. The second prototype demo described in this chapter builds on the first demo, adds the additional envisioned trust-building enhancements and introduces some necessary modifications.

5.1 The Auto-Choice-Advisor Prototype Website

5.1.1 Web Technology Needs

In order to build the question answering functionality and the Virtual Engineer trigger mechanism and logic, the prototype website needed to be changed to a more dynamic implementation. It was necessary to enable parameters, such as user responses, to be communicated from the user's browser back to a server which can in turn handle processing them, storing them as needed in a database and deciding on the next page to be served to the user. This organization of the system is summarized in figure 5.1 below.

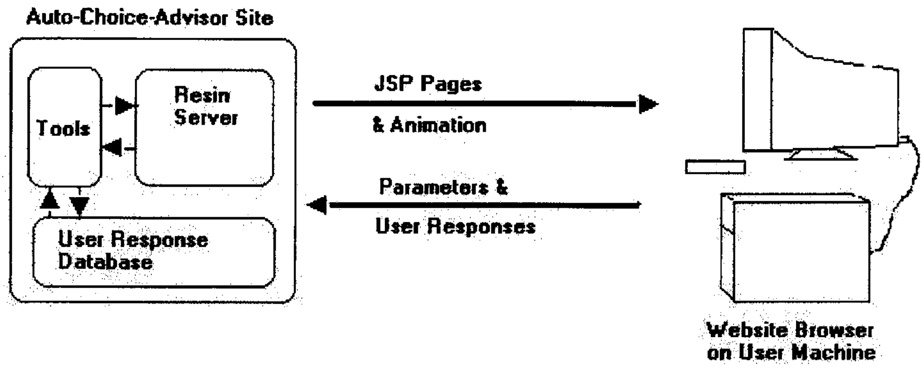

Figure 5.1: Dynamic Pages

The first step was to setup a server that would serve the website's pages to clients, users of the site. A Resin server was setup for this purpose. Next a Microsoft Access database was setup on the server machine to record the user responses. The static HTML pages were then changed to Java Server Pages (JSP) the contents of which are generated dynamically by the Resin server. The choice of the JSP and Resin technologies over other dynamic page options was due to their availability as freeware. At the same time, in order to manage user responses on the user's browser and consequently pass them to the server, the pages were powered with Java Applets that implement the input forms and buttons in the survey and perform the necessary processing of the responses prior to passing them to the server.

5.1.2 Website Design and Implementation

The website was laid out in a very similar way to the real Auto-Choice-Advisor site. The general organization of the site is outlined in figure 5.2 below. At the first page, the animated advisor Louis introduces himself and explains what the site is about. The user then proceeds to navigate, with Louis' help, through the survey answering questions. When the user begins answering the survey, the server adds a cookie to the user's browser that contains a unique user ID. The ID is checked whenever data is shared with the server and the server uses it to store and retrieve user data and responses from the database. Whenever the user is done with a page, any responses the user provided are passed to the next page the user visits with the help of the Java Applets. The next page in turn handles passing them to the server.

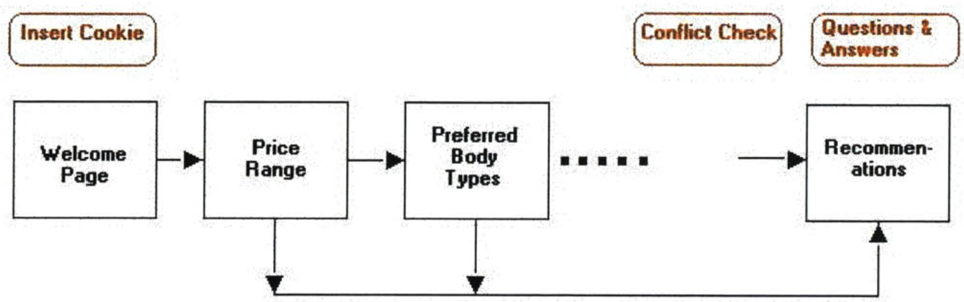

Figure 5.2: Auto-Choice-Advisor Demo Site Organization

Similar to the current Auto-Choice-Advisor site, the user can view the recommended vehicles at any point during the visit. For the purpose of the demo, the recommendation page remained a static page with a static set of recommended vehicles. This is simply because implementing a dynamic recommendation page would be unnecessarily replicating a functionality that is already available on the Auto-Choice-Advisor site and does not help in demonstrating the added value of a trusted animated advisor. On the recommendation page, Louis offers the user to ask any questions he might have about the recommended cars. This is where the question answering unit was integrated in the site.

When the user is done with the survey, a set of logic rules examine the responses of the user on the server and check for any unsatisfiable requirements. If they exist, the Virtual Engineer is launched and further dialogue with the user about their preferences takes place. If not, the user proceeds to the recommendation page. The reason for launching the engineer at the end of the visit only, if necessary, is due to usability considerations. The engineer should not interrupt the work of the virtual advisor who is working on walking the visitor through the site. This could confuse the visitors and be destructive to their visit experience. Furthermore, the visitors may choose not to have any further dialogues with the engineer and might prefer to proceed to the recommendations, their goal for visiting the site.

5.2 The Virtual Advisor

5.2.1 The Animation Technology

The animated advisor, Louis, was implemented using LifeFX technology which is composed of an ActiveX control player that runs a stand-in persona in the user's web browser based on compressed expression files sent by the server. The expression files describe the voice, the facial gestures and the head movements the animated persona should perform. This animation technology works with Internet Explorer only and requires downloading a plug-in.

While animation technology can be coupled with voice synthesis technology to enable animated advisors to speak any dynamically generated text on the fly, the LifeFX technology available to the team at the time of developing this prototype required the pre-recording of any voice or expression files needed for the animation. This constraint imposed some limitations on the implementation of the virtual advisor demo. However, if the proposed next generation of Auto-Choice-Advisor is launched on the real Auto-Choice-Advisor website, this constraint can easily be overcome by obtaining the extra technologies needed.

5.2.2 Interacting with the Visitors: Question Answering Unit

Natural Language Processing Needs and Issues

In order to run the animation with a question answering capability, it has to be powered by a back-end natural language engine. The engine would need to parse the user's text input and use it, together with natural language processing logic to extract the needed information from a knowledge database, possibly analyze it and then later generate the English language output needed to communicate the information to the user.

Natural language processing is a hard unsolved problem being explored by the artificial intelligence community. Building a system that can analyze and respond to the questions of Auto-Choice-Advisor visitors would be a doable but difficult and very time consuming process. For the simplest types of questions about vehicle specifications, such as inquiring about the maximum towing capacity of the GMC Sierra, natural language techniques are available that can handle that. It is also possible to build a system that responds to questions requiring comparisons based on specifications, such as inquiring about the car with the smallest turning circle among the recommendations. More intelligent question answering systems would be a lot harder to build specially when the context of a question is not clear from the way the question is phrased or when harder decision making is needed. An example would be if the user asks which car is the best value for money.

Designing and implementing a natural language processing engine for the Auto-Choice-Advisor demo would be highly time demanding and difficult to achieve within the time frame of the project. There were multiple ways to go around the bottleneck of natural language processing. The simplest and quite unsophisticated way is to define every possible question that can come up and map it to the necessary answer. However, this is a tedious process that does not generate good results even if all possible questions are answered. Moreover, in addition to the fact that the questions and answers would need to be predefined for every single vehicle, natural language processing would still be needed to detect the many different ways the same question can be asked. In short, the simplest way is quite difficult. A better alternative would be to borrow a ready off-the shelf question answering toolkit that understands the different ways a question can be asked. This would take care of the difficulty of detecting whether certain questions are simple paraphrases of one another. However, this approach does not eliminate the need to define the questions and answers for every single question, a difficulty which definitely leaves the time-cost issues practically irresolvable.

If the natural processing engine were to be designed and implemented from scratch for the real Auto-Choice-Advisor site, which is the only robust option if question answering were to be launched on the real site, the problem of defining the questions and answers for every single vehicle should no longer exist. This is because, given that a natural language engine is designed and implemented, the necessary queries to the vehicle database can be easily integrated with the natural language engine to analyze the visitors' questions and generate the answers dynamically by extracting any necessary information from the database.

Since, for the purpose of the second demo, the goal was to demonstrate the potential capability of an animated advisor, the focus of the demo was to answer questions about the predefined vehicle recommendation only. Using a ready off-the-shelf tool that can understand different ways a question can be asked was sufficient if all the needed questions and answers about the recommended vehicles were defined. Given the goals of the demo and the time-frame of the project, this was a sufficient solution. An on-line service, My[Q]Box was used to manage to the question base and to handle the back-end of the question answering.

My[Q]Box provides a web access to its tools via an account subscription. The questions and answers are fed into a knowledge base on the My[Q]Box site by the administrator of the account. My[Q]Box then provides a free website where queries to the knowledge base, the questions, can be input and the results, the answers, are displayed. The next section will go over the design of this question answering demo.

Design and Implementation

In order to answer visitor questions via the animation, both the text of the response and the audio files needed to be retrieved so that the answer text is displayed and the answer is read by the animation. For this purpose, the answers to the pre-defined questions were fed into My[Q]Box in a format that includes both the text and the name of the relevant animation expression file. Figure 5.3 below shows an example of a question defined in M[Q]Box and how the answer, text and expression file name, were formatted. The AUTOCHOICE tags on either side are added for the purpose of facilitating parsing the My[Q]Box HTML page, which contains the answer, and retrieving the desired parts.

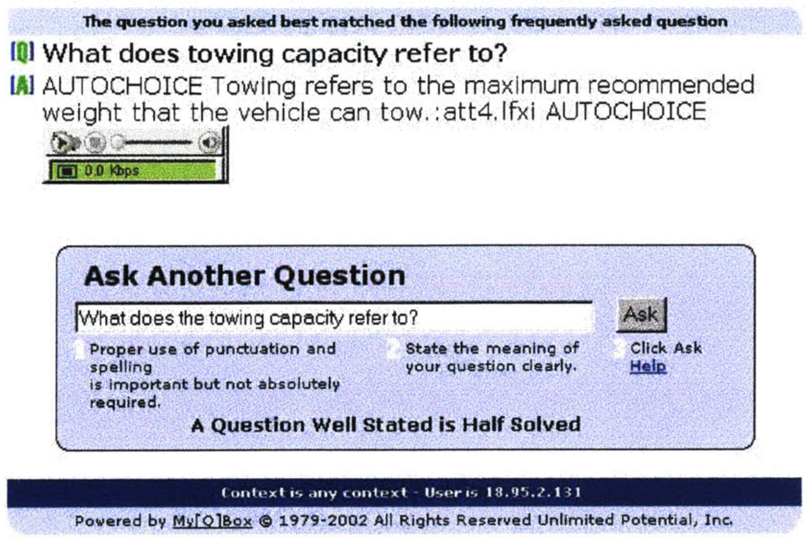

Figure 5.3: Specifying a Question and Its Answer in My[Q]Box

On the Auto-Choice Advisor demo site, the visitor's question is submitted via a web form on the page that has the recommended vehicles, the consideration page. This page is then reloaded and the text is passed, via the demo site server, to the My[Q]Box server which in turn decides on the answer and returns an HTML page containing the response. The Auto-Choice-Advisor demo server then parses the HTML page and extracts the text of the answer and the name of the animation expression file. It does so by extracting the text that lies between the two AUTOCHOICE tags and separating it around the ":" delimiter. The consideration page is then re-displayed showing the text of the response in the center of the page and the animation speaks the relevant audio response. Figure 5.4 bellow summarizes the design of the question answering unit.

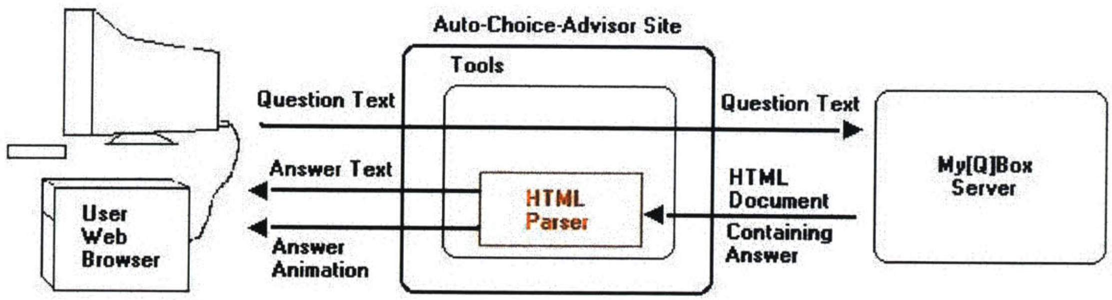

Figure 5.4: The Question-Answering Unit

5.3 The Virtual Engineer

5.3.1 Design Overview

The virtual engineer functionality, prototyped in this project, is a reactive rule-based system. The knowledge in the engineer is built in as a set of facts about the vehicles in the market as well as rules describing the situations in which customer requirements can be impossible given the market. At the end of a visit to the Auto-Choice-Advisor demo site, this functionality carries out a logic check to determine whether the visitor specified any impossible requirements. The rules are checked and applied to what is known about the market and what is known about the user's preferences, from the user's responses. If a rule is asserted, a conflict is identified in the user's responses and the virtual engineer mechanism is triggered and further dialogue with the visitor begins.

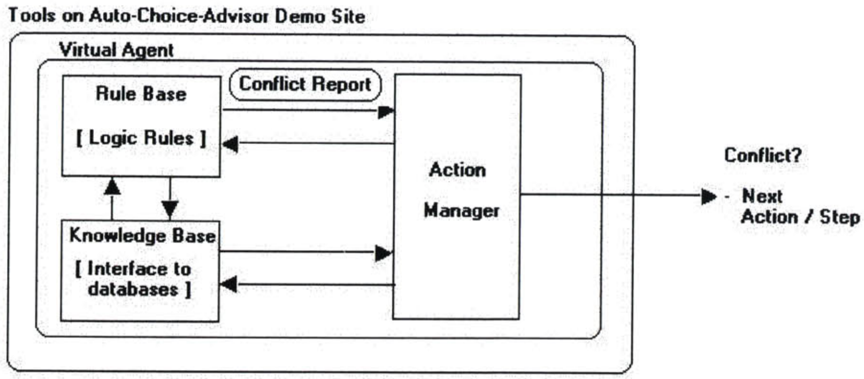

Figure 5.5: Rule-Based Reactive System

Figure 5.5 outlines the software organization of the engineer functionality. A Virtual Agent module encapsulates all of the built modules. A knowledge base is constructed on top of a local database on the server side that contains the relevant market data. This database stores knowledge about the vehicles in the market such as the towing capacities of these vehicles. Moreover, the knowledge base retrieves the user's responses, using the user's cookie ID, from another database, the database that listens and records the responses mentioned in section 5.1.1. A Rule Base module contains the rules that are checked against the market knowledge and user responses in the knowledge base. For every conflict that the Rule Base detects, a Conflict Report object is constructed. The Conflict Report contains all the details about the conflict, such as the type and the respective dialogue trace, which are needed for an Action Manager to determine its priority among all identified conflicts.

5.3.2 Prototype Module Specifications

The Virtual Agent

A VirtualAgent object is instantiated on the server when a visitor reaches the end of the visit to Auto-Choice-Advisor. It is constructed with an instance of the KnowledgeBase and an instance of RuleBase. It then starts the logic check, performed by the RuleBase, and then uses an ActionManager instance to handle the decision making on the order in which conflicts, described as ConflictReport objects, should be handled. For the purpose of the demo, at most only one conflict was handled in any visit since it was sufficient to demonstrate how the system would work and also in order to avoid very lengthy dialogues that might be too demanding of the user.

The Knowledge Base

A KnowledgeBase object is instantiated whenever an interface is needed to read market data or user responses from the databases. For the purpose of the demo, only a rough small subset of the market knowledge was built into the knowledge base. Specifically, the focus was the vehicles' towing capacities, hauling capacities and maneuverability. Furthermore, the focus was mainly on trucks. This was sufficient to demonstrate the workings and benefits of the virtual engineer functionality.

The Rule Base

An instance of RuleBase is created at the end of a user's visit to the Auto-Choice-Advisor demo site in order to perform the logic checks that decide whether the user responses are satisfiable by the vehicles available in the market. The rules built into the RuleBase for the demo are based on the towing capacity, hauling capacity and maneuverability of vehicles. First, a set of rules look at the user's overall car type selections and the towing, hauling and maneuverability requirements that the user specified. These rules try to determine whether the car selections match the set specifications. A second set of rules then determine whether the user indicated a most desirable vehicle type and, if so, whether the most preferred vehicle type satisfies the set requirements. In the Auto-Choice-Advisor demo the general rules were roughly estimated based on a quick survey of general towing, hauling and turning abilities of different vehicle types. The most-preferred vehicle type rules were setup for trucks only. Appendix II summarizes the rules that were implemented in the virtual engineer unit. Whenever a rule is asserted indicating the presence of a conflict a ConflictReport instance is created to archive the details of the conflict and then stored to be later passed to the ActionManager together with other ConflictReport instances for other conflicts.

The Conflict Report

The ConflictReport, instantiated whenever a conflict needs to be recorded, archives all the necessary details about the conflict. This includes the conflict type, the text and animation expression file name that can be used to inform the user about the conflict as well as other information that might need to be passed to the ActionManager about the conflict. This could include a list of the user's selected vehicles and their maximum towing capacities for example.

The Action Manager

An ActionManager instance examines the record of conflicts identified in a user's response at the end of the visit and determines the best way to address them. For the purpose of the demo, a simple implementation that gives priority to specific rules, related to trucks, over the general conflicts, involving all selected vehicles, was used.

5.3.3 Rule-Triggered Dialogues: Example

Figures 5.6 through 5.8 below show a sample scenario of a conflict triggered dialogue between the virtual engineer and the visitor. The user has specified that he needs a full size truck that can make tight turns. Full-size trucks are not very maneuverable and so a conflict is detected and the virtual engineer is launched.

Figure 5.6: A Conflict is Detected

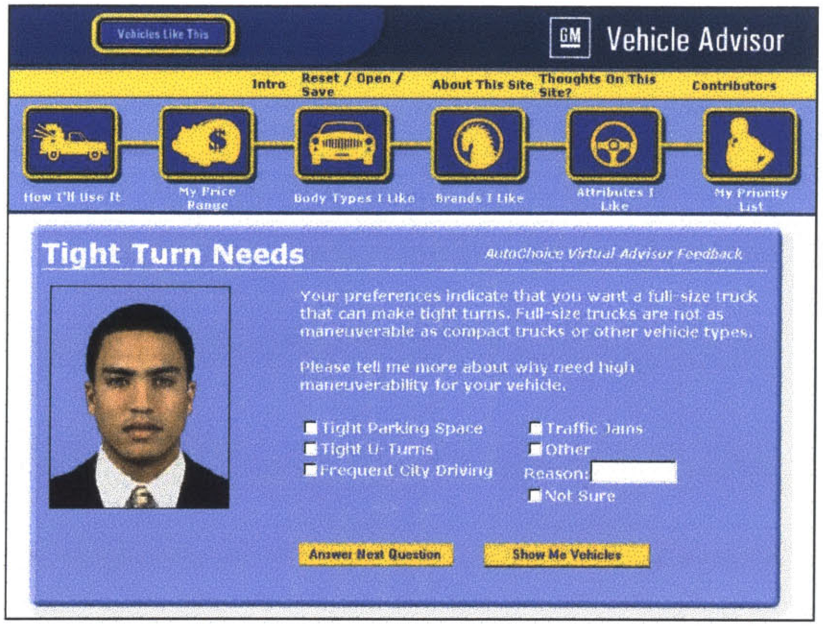

Figure 5.7: Inquiring About the Need for High Maneuverability

Figure 5.8: Inquiring About the Reason for the Truck Type Preference

Chapter 6: Summary and Future Work

This document presented an overview of the emergence of a new trust-based marketing strategy. It also documented the results and findings of the sensitivity tests that checked the internal validity of the "listening-in" approach, which can be a very beneficial approach in designing new products and bringing them to the market. Lastly, this document presented the design and implementation of a trusted animated advisor, Louis, that was prototyped on a test version of the Auto-Choice-Advisor website.

The animated advisor demo was presented at a meeting with customer relations management (CRM) executives at the General Motors headquarters. The goal of the meeting was to discuss a future panel-based research project managed by GM and MIT to study the effects of trust on customer satisfaction, behavior and, of course, actions such as making a purchase. The goal is also to study the effects of trust-building factors across all marketing channels. An animated talking-head advisor for the Auto-Choice-Advisor site was on the agenda for discussion an is under consideration to be included as a component of this future study.

References

[1] Glen L. Urban, "The Trust Imperative," March 2003, MIT Sloan School of Management 2003

[2] Stanley W. Cheung, "The Design and Evaluation of a Virtual Trusted Advisor," M.Eng. Thesis, MIT 2002.

[3] Fareena Sultan, Glen L. Urban, Venkatesh Shankar, Iakov Y. Bart, December 2002, MIT-SSM working paper, MIT 2002.

[4] Glen L. Urban, John R. Hauser, ""Listening In" to Find Unmet Customer Needs and Solutions," January 2003, MIT-SSM 2003.

[5] Hoi Wai Thomas Cheng, "Identifying Customers' Unmet Needs Using a Virtual Advisor and Engineer," M.Eng. Thesis, MIT 2001.

[6] Http://www.AutoChoiceAdvisor.com

Appendix I: "Listening-In" Monte Carlo Simulations

1. Profile Descriptions for the Simulation Tests

Each profile is associated with a standard set of Trucktown answers. Below is a listing of the default answers to the survey. Each profile will have different answers for a subset of these questions.

Standard Answers

1.	Compact/Full	= not sure
2a.	Number of Passengers	= 3
2b.	Number of Front Passengers	= 1
2c.	Easy Rear Entry	= 3
3,4.	Construction	= no
	Towing	= no
	Hauling	= no
	Offroad	= no
	Snow Plowing	= no
6.	Budget	= >32K
7.	Brand	= All no
8a.	Drivetrain	= either
8b.	WheelDrive	= either
9.	Style	= not sure
10.	Height	= under 6 feet
11.	Bed	= not sure
12.	Cylinders	= not sure (neither 4,6,8,10,diesel)
13.	Big & Comfort	= 3
14.	Maneuverability	= 3

Profile 1

Cylinders	= 4
Compact / Full	= Compact
Towing	= Yes
Hauling	= Yes

Profile 2

Style	= Sporty
Compact / Full	= Full
Bed	= Short
Cylinders	= 8
Big & Comfort	= 5

Profile 3

Style	= Conventional
Construction	= Yes
Cylinders	= Diesel
Big & Comfort	= 1

Profile 4

Style	= Rugged & Sporty
Compact / Full	= Compact
Bed	= Extra Short
Height	= 6-6'5
Cylinders	= 4

Profile 5

Style	= Conventional & Sporty
Towing	= Yes
Bed	= Short
Cylinders	= 10
Big & Comfort	= 1

Profile 6

Style	= Conventional
Maneuverability	= 5
Bed	= Long
Cylinders	= 6
Big & Comfort	= 4

Profile 7

Compact / Full	= Compact

Profile 8

Compact / Full	= Full
Construction	= Yes
Towing	= Yes
Hauling	= Yes

Profile 9

Compact / Full	= Full

2. Sensitivity to Trigger Level Runs:

These runs are based on 5-point constant-sum error and 10% nominal category error.

T = 0	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Cluster10	Total
Profile 1	**340**	1	0	1	0	0	158	0	0	0	500
Profile 2	0	**432**	0	0	0	0	68	0	0	0	500
Profile 3	0	0	**403**	0	0	0	97	0	0	0	500
Profile 4	1	0	0	**346**	0	0	153	0	0	0	500
Profile 5	0	28	0	0	**337**	0	135	0	0	0	500
Profile 6	0	1	0	0	0	**343**	95	44	17	0	500
7, 8 & 9	1	0	2	1	0	0	**1496**	0	0	0	1500
Rate	**82.16%**										

T=0.000025	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Cluster10	Total
Profile 1	**418**	0	0	1	0	0	81	0	0	0	500
Profile 2	1	**422**	0	0	0	0	77	0	0	0	500
Profile 3	0	0	**401**	0	0	0	99	0	0	0	500
Profile 4	1	0	0	**346**	0	0	153	0	0	0	500
Profile 5	3	27	0	0	**336**	0	119	15	0	0	500
Profile 6	0	2	0	0	0	**346**	92	0	43	17	500
7, 8 & 9	43	0	2	1	0	0	**1454**	0	0	0	1500
Rate	**82.73%**										

T=0.00005	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Cluster10	Total
Profile 1	**418**	0	0	1	0	0	81	0	0	0	500
Profile 2	1	**422**	0	0	0	0	77	0	0	0	500
Profile 3	0	0	**401**	0	0	0	99	0	0	0	500
Profile 4	1	0	0	**346**	0	0	153	0	0	0	500
Profile 5	3	27	0	0	**336**	0	119	15	0	0	500
Profile 6	0	2	0	0	0	**346**	92	0	43	17	500
7, 8 & 9	43	0	2	1	0	0	**1454**	0	0	0	1500
Rate	**82.73%**										

T=0.000075	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Cluster10	Total
Profile 1	**418**	1	0	1	0	0	80	0	0	0	500
Profile 2	1	**420**	0	0	0	0	79	0	0	0	500
Profile 3	0	1	**401**	0	0	0	98	0	0	0	500
Profile 4	1	21	0	**346**	0	0	132	0	0	0	500
Profile 5	3	3	0	0	**337**	0	142	15	0	0	500
Profile 6	0	0	0	0	0	**345**	95	0	43	17	500
7, 8 & 9	43	0	2	1	0	0	**1454**	0	0	0	1500
Rate	**82.69%**										

T=0.0001	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Cluster10	Total
Profile 1	**418**	1	0	1	0	0	80	0	0	0	500
Profile 2	1	**420**	0	0	0	0	79	0	0	0	500
Profile 3	0	1	**401**	0	0	0	98	0	0	0	500
Profile 4	1	21	0	**346**	0	0	132	0	0	0	500
Profile 5	3	3	0	0	**337**	0	142	15	0	0	500
Profile 6	0	0	0	0	0	**345**	95	0	43	17	500
7, 8 & 9	43	0	2	1	0	0	**1454**	0	0	0	1500
Rate	**82.69%**										

T=0.00015	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Cluster10	Total
Profile 1	**418**	1	0	1	0	0	80	0	0	0	500
Profile 2	1	**420**	0	0	0	0	79	0	0	0	500
Profile 3	0	1	**401**	0	0	0	98	0	0	0	500
Profile 4	1	21	0	**346**	0	0	132	0	0	0	500
Profile 5	3	3	0	0	**337**	0	142	15	0	0	500
Profile 6	0	0	0	0	0	**345**	95	0	43	17	500
7, 8 & 9	43	0	2	1	0	0	**1454**	0	0	0	1500
Rate	**82.69%**										

T=0.001	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Cluster10	Total
Profile 1	**427**	1	0	1	0	0	71	0	0	0	500
Profile 2	0	**415**	0	0	0	0	85	0	0	0	500
Profile 3	1	0	**399**	0	0	0	100	0	0	0	500
Profile 4	3	1	0	**346**	0	0	150	0	0	0	500
Profile 5	1	18	0	0	**336**	0	145	0	0	0	500
Profile 6	0	0	0	0	0	**345**	95	43	17	0	500
7, 8 & 9	43	0	2	1	0	0	**1454**	0	0	0	1500
Rate	**82.71%**										

T=0.01	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Cluster10	Total
Profile 1	263	0	0	0	0	0	237	0	0	0	500
Profile 2	0	0	0	0	0	0	500	0	0	0	500
Profile 3	0	0	308	0	0	0	192	0	0	0	500
Profile 4	0	17	0	0	0	23	460	0	0	0	500
Profile 5	0	0	0	239	0	0	236	25	0	0	500
Profile 6	0	0	0	0	241	0	233	0	26	0	500
7, 8 & 9	0	0	0	0	0	0	1500	0	0	0	1500
Rate	**56.69%**										

T=0.1	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Cluster10	Total
Profile 1	1	1	1	1	1	1	491	1	2	0	500
Profile 2	0	0	0	0	0	0	500	0	0	0	500
Profile 3	0	0	0	0	0	0	500	0	0	0	500
Profile 4	0	0	0	0	0	0	500	0	0	0	500
Profile 5	0	0	0	0	0	0	500	0	0	0	500
Profile 6	0	0	0	0	0	0	500	0	0	0	500
7, 8 & 9	0	0	0	0	0	0	1500	0	0	0	1500
Rate	**33.33%**										

3. Response Error Sensitivity Runs

A	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Total
Profile 1	**497**	0	0	0	0	0	0	1	2	500
Profile 2	0	**500**	0	0	0	0	0	0	0	500
Profile 3	0	0	**500**	0	0	0	0	0	0	500
Profile 4	0	0	0	**500**	0	0	0	0	0	500
Profile 5	0	0	0	0	**500**	0	0	0	0	500
Profile 6	0	0	0	0	0	**500**	0	0	0	500
7, 8 & 9	0	0	0	0	0	0	**1500**	0	0	1500
Rate	**99.93%**									

B	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Total
Profile 1	**365**	0	1	1	0	0	133	0	0	500
Profile 2	0	**447**	0	0	0	0	53	0	0	500
Profile 3	0	0	**410**	0	0	0	90	0	0	500
Profile 4	5	0	0	**306**	0	0	136	53	0	500
Profile 5	0	16	0	1	**357**	0	126	0	0	500
Profile 6	0	5	1	0	0	**358**	110	0	26	500
7, 8 & 9	2	0	10	0	0	1	**1486**	1	0	1500
Rate	**82.87%**									

C	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Total
Profile 1	**231**	2	0	1	0	0	235	31	0	500
Profile 2	3	**291**	1	0	0	0	203	2	0	500
Profile 3	8	2	**306**	0	0	0	183	1	0	500
Profile 4	6	5	0	**219**	0	0	145	125	0	500
Profile 5	5	16	4	0	**272**	0	202	1	0	500
Profile 6	1	0	3	0	0	**237**	220	1	38	500
7, 8 & 9	132	43	26	0	0	1	**1214**	84	0	1500
Rate	**61.56%**									

D	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Total
Profile 1	**497**	0	0	0	0	0	0	1	2	500
Profile 2	0	**500**	0	0	0	0	0	0	0	500
Profile 3	0	0	**500**	0	0	0	0	0	0	500
Profile 4	0	0	0	**500**	0	0	0	0	0	500
Profile 5	0	0	0	0	**499**	1	0	0	0	500
Profile 6	0	0	0	0	0	**499**	1	0	0	500
7, 8 & 9	0	0	0	0	0	0	**1499**	0	0	1499
Rate	**99.87%**									

E	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Total
Profile 1	**418**	0	0	1	0	0	81	0	0	500
Profile 2	1	**422**	0	0	0	0	77	0	0	500
Profile 3	0	0	**401**	0	0	0	99	0	0	500
Profile 4	1	0	0	**346**	0	0	153	0	0	500
Profile 5	3	27	0	0	**336**	0	134	0	0	500
Profile 6	0	2	0	0	0	**346**	92	43	17	500
7, 8 & 9	43	0	2	1	0	0	**1454**	0	0	1500
Rate	**82.73%**									

F	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Total
Profile 1	**290**	51	0	1	0	0	147	11	0	500
Profile 2	7	0	0	0	0	1	491	1	0	500
Profile 3	7	0	**309**	0	0	0	182	2	0	500
Profile 4	11	9	0	**237**	0	0	140	103	0	500
Profile 5	7	0	10	0	**179**	1	303	0	0	500
Profile 6	4	0	0	0	0	**230**	212	1	53	500
7, 8 & 9	194	0	18	0	0	2	**1230**	56	0	1500
Rate	**55.00%**									

G	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Total
Profile 1	**497**	0	0	0	0	0	0	1	2	500
Profile 2	0	**499**	0	0	0	0	1	0	0	500
Profile 3	0	0	**500**	0	0	0	0	0	0	500
Profile 4	0	0	0	**500**	0	0	0	0	0	500
Profile 5	0	0	0	0	**500**	0	0	0	0	500
Profile 6	0	0	0	0	0	**500**	0	0	0	500
7, 8 & 9	0	0	0	0	0	0	**1500**	0	0	1500
Rate	**99.91%**									

H	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Total
Profile 1	**360**	0	0	0	0	0	139	1	0	500
Profile 2	0	**414**	0	0	0	0	86	0	0	500
Profile 3	0	0	**401**	0	0	0	99	0	0	500
Profile 4	2	1	0	**334**	0	0	116	47	0	500
Profile 5	1	1	6	0	**332**	0	160	0	0	500
Profile 6	0	1	0	0	0	**346**	115	0	38	500
7, 8 & 9	1	2	1	0	0	0	**1494**	2	0	1500
Rate	**81.80%**									

I	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8	Cluster9	Total
Profile 1	**300**	27	0	2	0	0	171	0	0	500
Profile 2	0	0	3	0	0	0	497	0	0	500
Profile 3	1	2	**321**	0	0	0	175	1	0	500
Profile 4	11	49	1	**213**	0	0	226	0	0	500
Profile 5	1	0	5	0	**167**	0	293	0	34	500
Profile 6	2	0	2	0	0	**253**	186	57	0	500
7, 8 & 9	86	91	22	0	0	0	**1301**	0	0	1500
Rate	**56.78%**									

4. Run E Cluster Composition:

Run E had 5-Point constant-sum response error and 10% nominal-category response error. The statistics below show that every meaningful output cluster corresponded to one of the customer profiles. Each major cluster contained the expected conflicts for more than 50% of its members. No other unexpected conflicts appeared for more than 15.08% of the members of any major cluster. 15.08% is way below the set criteria for mis-identification, 50% representation in a major size cluster.

Cluster 1

Total	***466***
Conf5	95.92%
Conf6	82.40%
Conf76	77.25%
Conf77	72.32%
Max Other	9.44%

Cluster 2

Total	***451***
Conf19	79.60%
Conf51	86.03%
Conf98	78.49%
Conf100	84.48%
Conf156	82.71%
Conf 160	94.46%
Max Other	15.08%

Cluster 3

Total	***403***
Conf12	92.56%
Conf112	86.35%
Conf122	99.50%
Conf135	93.05%
Max Other	12.66%

Cluster 4

Total	***348***
Conf23	99.43%
Conf49	86.78%
Conf63	99.71%
Conf67	87.07%
Conf80	97.70%
Conf83	97.99%
Max Other	11.49%

Cluster 5

Total	***336***
Conf15	95.54%
Conf52	89.58%
Conf105	96.73%
Conf108	90.48%
Conf120	95.83%
Conf134	97.02%
Max Other	11.61%

Cluster 6

Total	***346***
Conf92	84.39%
Conf154	92.20%
Conf162	91.04%
Conf182	92.49%
Conf188	100.00%
Conf201	91.62%
Conf207	99.13%
Max Other	10.98%

Cluster 7

Total	***2090***
Max Other	6.65%

Cluster 8

Total	***43***
Conf12	97.67%
Conf86	20.93%
Conf88	97.67%
Conf92	81.40%
Conf154	97.67%
Conf162	97.67%
Conf176	20.93%
Conf178	100.00%
Conf182	83.72%
Conf188	100.00%
Conf191	20.93%
Conf193	100.00%
Conf201	83.72%
Conf207	100.00%
Max Other	9.30%

Cluster 9

Total	***17***
Conf64	100.00%
Conf70	100.00%
Conf87	23.53%
Conf92	82.35%
Conf93	88.24%
Conf94	88.24%
Conf154	100.00%
Conf162	88.24%
Cond177	23.53%
Conf182	94.12%
Conf183	100.00%
Conf184	100.00%
Conf188	100.00%
Conf192	23.53%
Conf201	94.12%
Conf202	100.00%
Conf203	100.00%
Conf207	100.00%
Max Other	17.65%

5. False Positive and False Negative Statistics

Below is a sample set of statistics that show the percentages of false positive and false negative conflicts in each of the clusters. False negative occurs when the an expected conflict does not appear for a member of the cluster and similarly false positive is when a member of a cluster has a conflict that is not expected to be present in that cluster.

Profile or Cluster	Expected Conflicts	# of False Negative	False Neg / (Ex # Confs)	False Neg / (# obs Confs)	# of False Positive	False Pos / (# obs Confs)	Total Observed Conflicts
1	4	372	18.6%	15.9%	715	30.5%	2343
2	6	564	18.8%	16.7%	945	28.0%	3381
3	4	336	16.8%	12.8%	958	36.5%	2622
4	6	558	18.6%	15.8%	1086	30.8%	3528
5	6	570	19.0%	16.1%	1108	31.3%	3538
6	7	564	16.1%	12.8%	1464	33.3%	4400
7	0	0	N/A	0%	438	100%	438
8	0	0	N/A	0%	600	100%	600
9	0	0	N/A	0%	246	100%	246

Appendix II: Auto-Choice-Advisor Demo

if Towing **AND** Subcompact car **then** conflict 1		**if** Towing **AND** very heavy **AND** Compact Truck **then** conflict 7
if Towing **AND** compact car **then** conflict 2	**if** Towing **AND** heavy **AND** Mini Van **then** conflict 8h	**if** Towing **AND** very heavy **AND** Mini Van **then** conflict 8v
if Towing **AND** midsize car **then** conflict 3		**if** Towing **AND** very heavy **AND** Full Size Van **then** conflict 9
if Towing **AND** large car **then** conflict 4	**if** Towing **AND** heavy **AND** Mini Sport Utility 2D **then** conflict 10h	**if** Towing **AND** very heavy **AND** Mini Sport Utility 2D **then** onflict 10v
if Towing **AND** Convertible Car **then** conflict 5	**if** Towing **AND** heavy **AND** Mini Sport Utility 4D **then** conflict 11h	**if** Towing **AND** very heavy **AND** Mini Sport Utility 4D **then** conflict 11v
if Towing **AND** Sports Car **then** conflict 6		**if** Towing **AND** very heavy **AND** Medium Sport Utility 2D **then** conflict 12
		if Towing **AND** very heavy **AND** Medium Sport Utility 4D **then** conflict 13
if Towing **AND** moderate **AND** Small Wagon **then** conflict 14m	**if** Towing **AND** heavy **AND** Small Wagon **then** conflict 14h	**if** Towing **AND** very heavy **AND** Small Wagon **then** conflict 14v
if Towing **AND** moderate **AND** Mid/Large Wagon **then** conflict 15m	**if** Towing **AND** heavy **AND** Mid/Large Wagon **then** conflict 15h	**if** Towing **AND** very heavy **AND** Mid/Large Wagon **then** conflict 15v

Towing a Trailer: Logic Rules

if	Hauling large / heavy items
	AND
	Compact Truck
then	conflict 1

if	Hauling large / heavy items
	AND
	Mini Van
then	conflict 2

if	Hauling large / heavy items
	AND
	Mini Sport Utility 2D
then	conflict 3

if	Hauling large / heavy items
	AND
	Mini Sport Utility 4D
then	conflict 4

if	Hauling large / heavy items
	AND
	Convertible Car
then	conflict 5

if	Hauling large / heavy items
	AND
	Sports Car
then	conflict 6

if	Hauling large / heavy items
	AND
	Subcompact car
then	conflict 7

if	Hauling large / heavy items
	AND
	compact car
then	conflict 8

if	Hauling large / heavy items
	AND
	midsize car
then	conflict 9

if	Hauling large / heavy items
	AND
	large car
then	conflict 10

if	intermediate **OR** small turning angle
	AND
then	Full Truck
	conflict 1

if	intermediate **OR** small turning angle
	AND
then	Full Van
	conflict 2

Hauling and Turning Circle: Logic Rules